



# Overview of Model Assurance Levels (MALs) for Systems and Software Models

February 2020

Julie S Fant, Karen McShane, Vineet Velmurugan, Ronald Nussbaum, Robert Pettit, PhD  
Software Engineering Subdivision

Prepared for:

Senior Vice President, Engineering and Technology Group

Authorized by: Engineering and Technology Group

# Agenda



# Agenda



# ***Motivation for Model Assurance Levels (MALs)***



*How confident can I be in the models?*

*Are the models adding enough value and risk reduction to the program?*

*Are the models mature enough for the acquisition or development phase?*

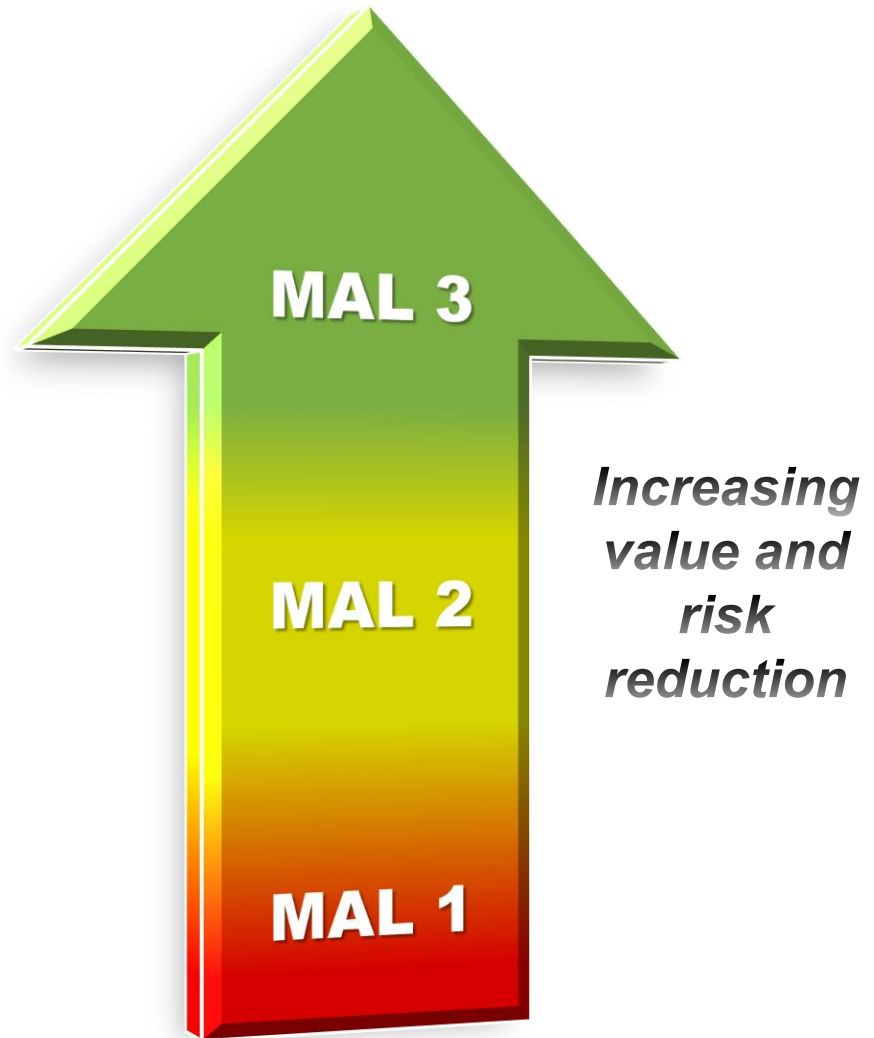
*How to talk about models to decisions makers?*



***Difficult for those without detailed knowledge of MBE to understand if models are providing the right value & risk reduction***

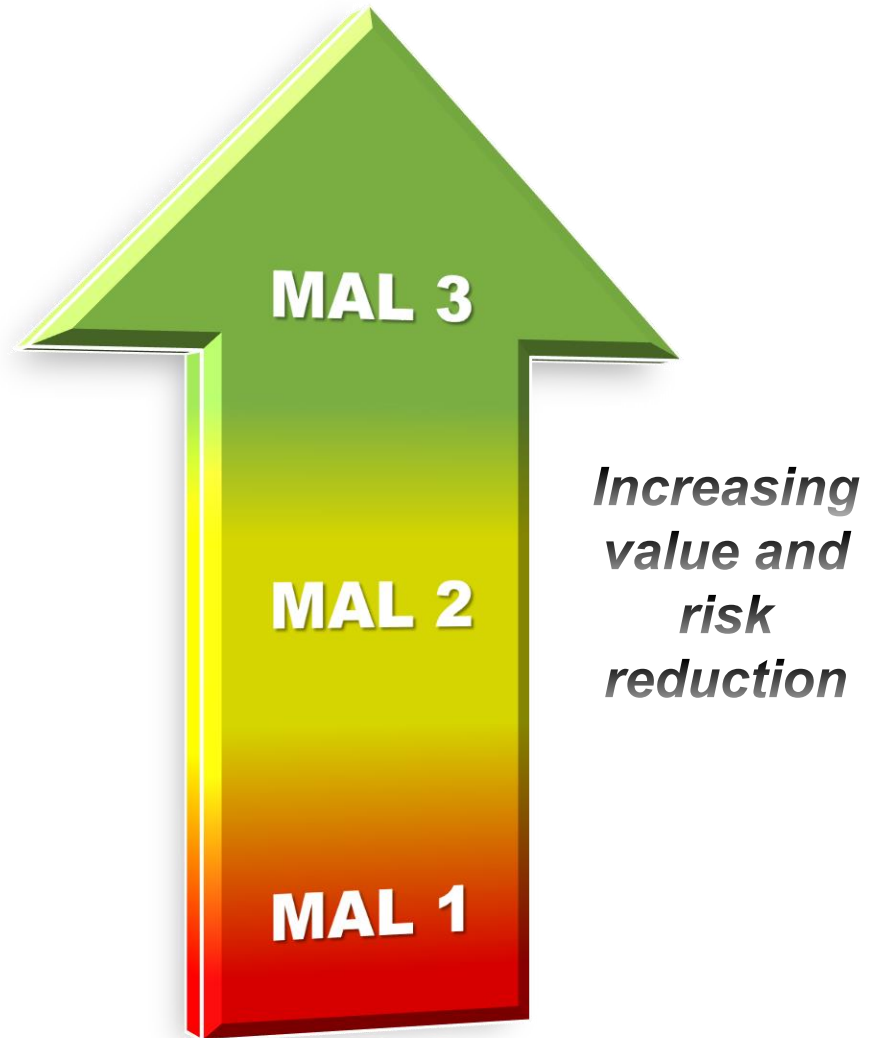
# What are Model Assurance Levels (MALs)

- **Model Assurance Level (MAL)**
  - *A measurement system for model value and quality*
  - *Based on a scale from 1-3*
    - Sub-levels developed to show incremental growth
  - *Inspired by NASA's TRLs*



# What are Model Assurance Levels (MALs)

- Model Assurance Level (MAL)
  - *A measurement system for model value and quality*
  - *Based on a scale from 1-3*
    - Sub-levels developed to show incremental growth
  - *Inspired by NASA's TRLs*
- **MAL Assessment Method**
  - *A repeatable and quantitative approach that examines the model(s) to determine the MAL*
  - *Identifies risk areas related to the models*



# ***What MAL Score Do You Want?***



***Risk is reduced  
with increasing  
MAL scores***



***Cost is increased  
with increasing  
MAL scores***

***Desired MAL score should be a balance of risk reduction and cost***



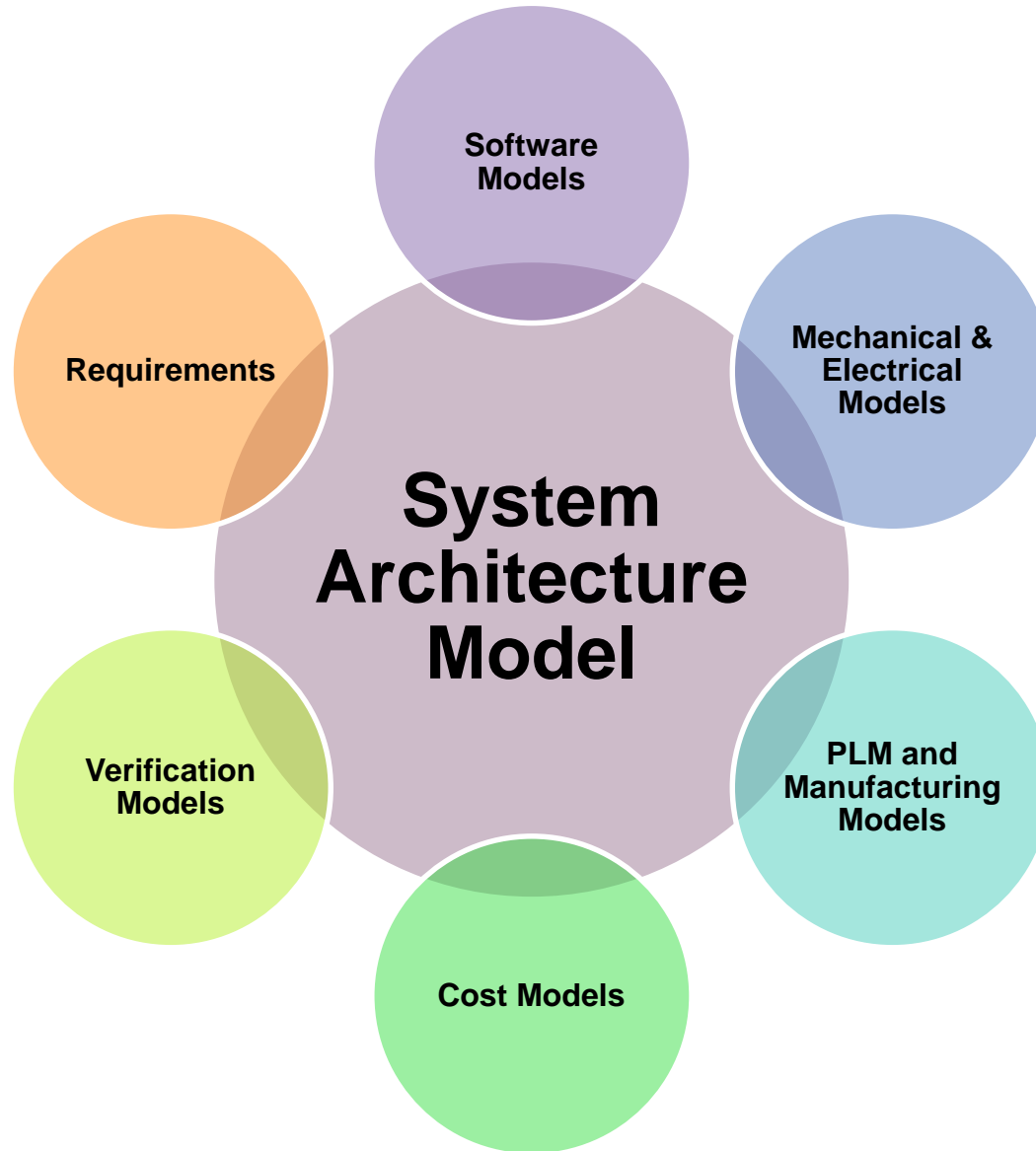


# Benefits of MALs

- **MALs are acquisition and development approach agnostic.**
  - *They can be determined and assessed regardless of acquisition structure or development lifecycle approach..*
- **MAL scores don't have to be universal across a program or project.**
  - *Models can be tailored for different purposes, so likewise, different domain models can have different desired MALs.*
- **MALs can be specified in acquisition language and proposals.**
  - *Using MALs in acquisition language or proposals can help set MBE expectations between government and contractor during proposal and contract award stages, as well as help avoid confusion on what types and the amount of content that will be developed.*
- **MAL assessment can be repeated to measure model development progress.**
  - *MAL assessments provide quantitative and repeatable results, therefore, they can be used to quantitatively assess model and quantitatively determine development progress.*

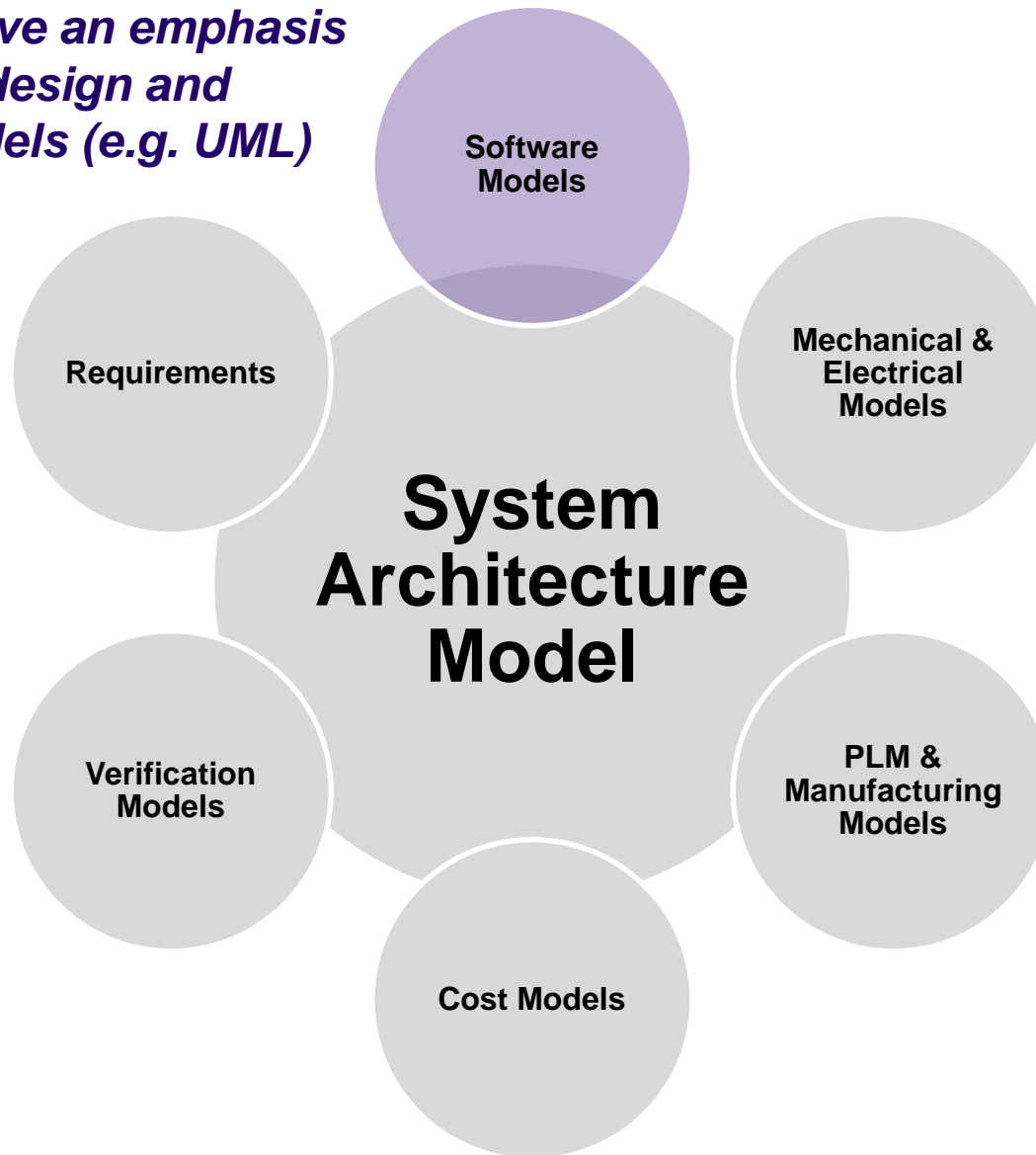


# ***MAL Types – Many types of models out there***



# MAL Types

*Software MALs have an emphasis on software design and architecture models (e.g. UML)*



# Software MAL Scale



MAL Level		Sub Level	Description
3	Advanced Models	3.8	Detailed Model that is incorporated into operation system
		3.7	Detailed Design Model use to drive implementation testing
		3.6	Completely Validated Complete Detailed Design Model
		3.5	Sparsely Validated Complete Detailed Design Model
		3.4	Complete Detailed Design Model
		3.3	Completely V&V Complete Analysis Model
		3.2	Sparse V&V Complete Analysis Model
		3.1	Complete Analysis Model
2	Basic Models	2.6	Completely V&V Basic Detailed Design Model
		2.5	Sparsely V&V Basic Detailed Design Model
		2.4	Basic Detailed Design Model
		2.3	Completely V&V Basic Analysis Model
		2.2	Sparsely V&V Basic Analysis Model
		2.1	Basic Analysis Model
1	Sparse Models	1.4	Sparsely Validated Sparse Detailed Design Model
		1.3	Sparse Detailed Design Model
		1.2	Sparsely V&V Space Analysis Model
		1.1	Sparse Analysis Model

Sparse = addresses some subset of software requirements

Basic = only address functional software requirements

Complete = addresses non-functional and functional software requirements

# Software MAL Scale



MAL Level		Model Risks for MAL 1.1	
3	Advanced Models	<p>Risk associated with the part of the software that is not modelled. In other words, you don't know what you don't know</p> <p>Risk associated with alternative paths and error conditions</p> <p>Risk that non-functional requirements will not be met</p> <p>Implementation risk since low level details are not included in the model</p>	
2	Basic Models	<p>Risk that the model isn't the correct model because model V&amp;V has not been performed</p>	
1	Sparse Models	2.2	Basic Analysis Model
		2.1	Analysis Model
		1.4	Fully Validated Sparse Detailed Design Model
		1.3	Sparse Detailed Design Model
		1.2	Sparsely V&V Space Analysis Model
		1.1	Sparse Analysis Model

# Software MAL Scale

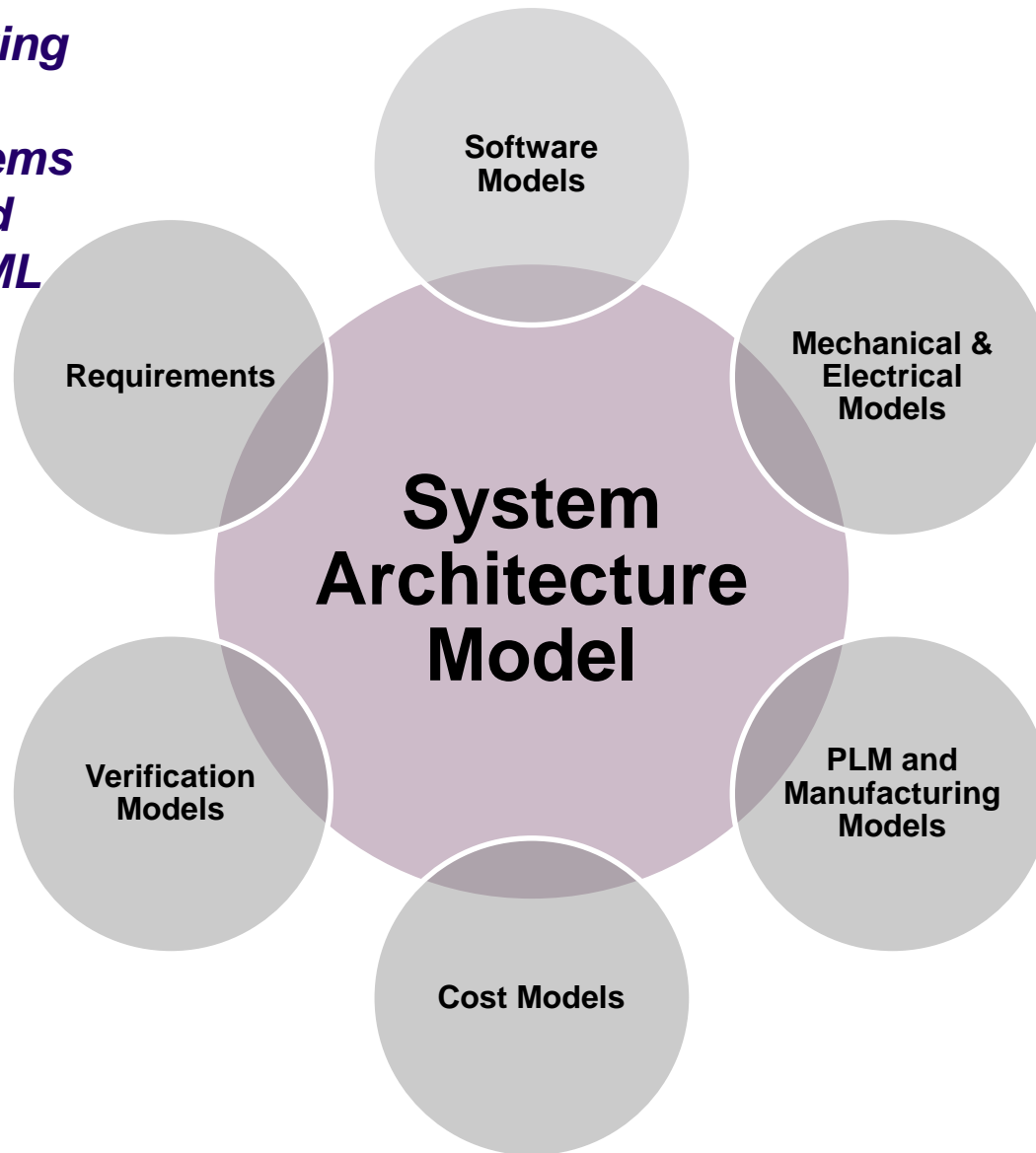


MAL Level		Sub Level	Description
3	Advanced Models	3.8	<b>Model Risks for MAL 1.4</b>  Risk associated with the part of the software that is not modelled. In other words, you don't know what you don't know  Risk associated with alternative paths and error conditions  Risk that non-functional requirements will not be met
		3.7	
		3.6	
		3.5	
		3.4	
		3.3	
		3.2	
		3.1	
2	Basic Models	2.6	
		2.5	
		2.4	
		2.3	Complete Space Analysis Model
		2.2	Sparsely V&V Basic Analysis Model
		2.1	Basic Analysis Model
1	Sparse Models	1.4	Sparsely Validated Sparse Detailed Design Model
		1.3	Sparse Detailed Design Model
		1.2	Sparsely V&V Space Analysis Model
		1.1	Sparse Analysis Model

# MAL Types



**Systems Engineering**  
MALs have an  
**emphasis on systems**  
**architecture and**  
**design (e.g. SysML**  
**model)**  
**It also examines**  
**if the model**  
**shares**  
**information with**  
**other domain**  
**models**



# Enterprise MAL Scale

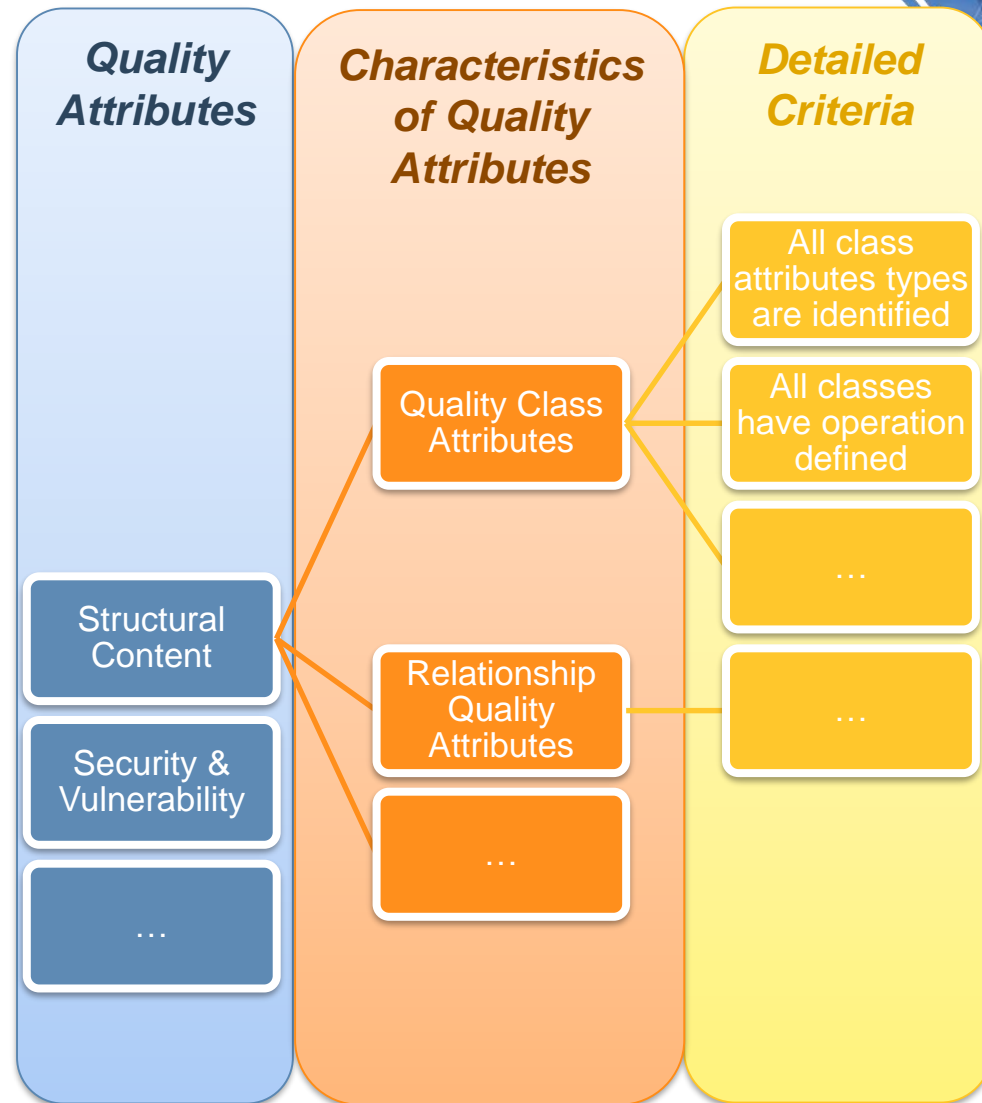


MAL Level		Sub Level	Description
3	Advanced Models	3.8	Advanced Allocated Model use to drive implementation testing
		3.7	Completely V&V Advanced Allocated Design Model
		3.5	Sparsely V&V Advanced Allocated Design Model
		3.4	Advanced Allocated Design Model
		3.3	Completely V&V Advanced Logical Design Model
		3.2	Sparsely V&V Advanced Logical Design Model
		3.1	Advanced Logical Design Model
2	Basic Models	2.9	Completely V&V Basic Allocated Design Model
		2.8	Sparsely V&V Basic Allocated Design Model
		2.7	Basic Allocated Design Model
		2.6	Completely V&V Basic Logical Design Model
		2.5	Sparsely V&V Basic Logical Design Model
		2.4	Basic Logical Design Model
		2.3	Completely V&V Basic Conceptual Design Model
		2.2	Sparsely V&V Basic Conceptual Design Model
		2.1	Basic Conceptual Design Model
1	Sparse Models	1.9	Sparsely Validated Sparse Physical Architecture Model
		1.8	Sparsely Validated Sparse Logical Architecture Model
		1.7	Sparsely Validated Sparse Conceptual Model
		1.3	Sparse Physical Architecture Model
		1.2	Sparse Logical Architecture Model
		1.1	Sparse Conceptual Model

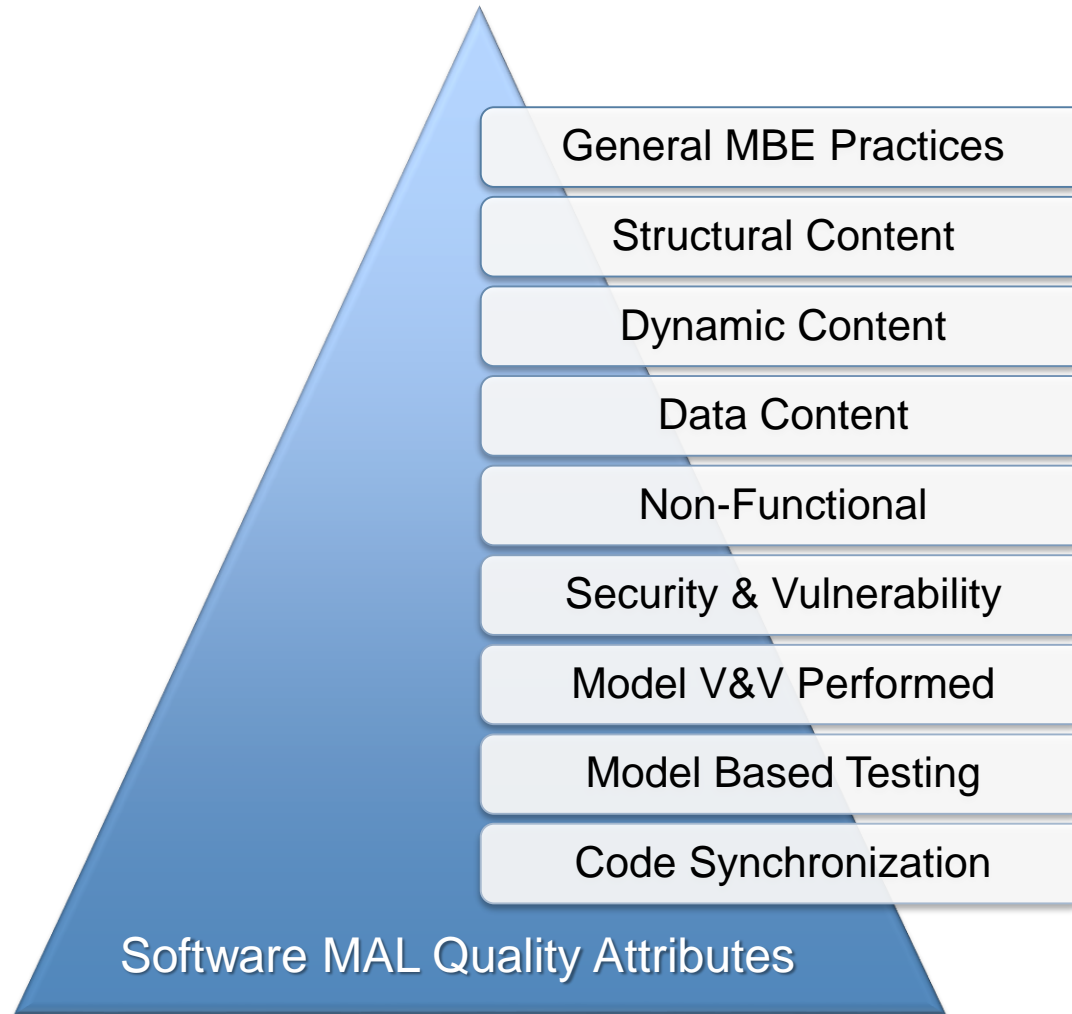


# How are MAL Scores Determined

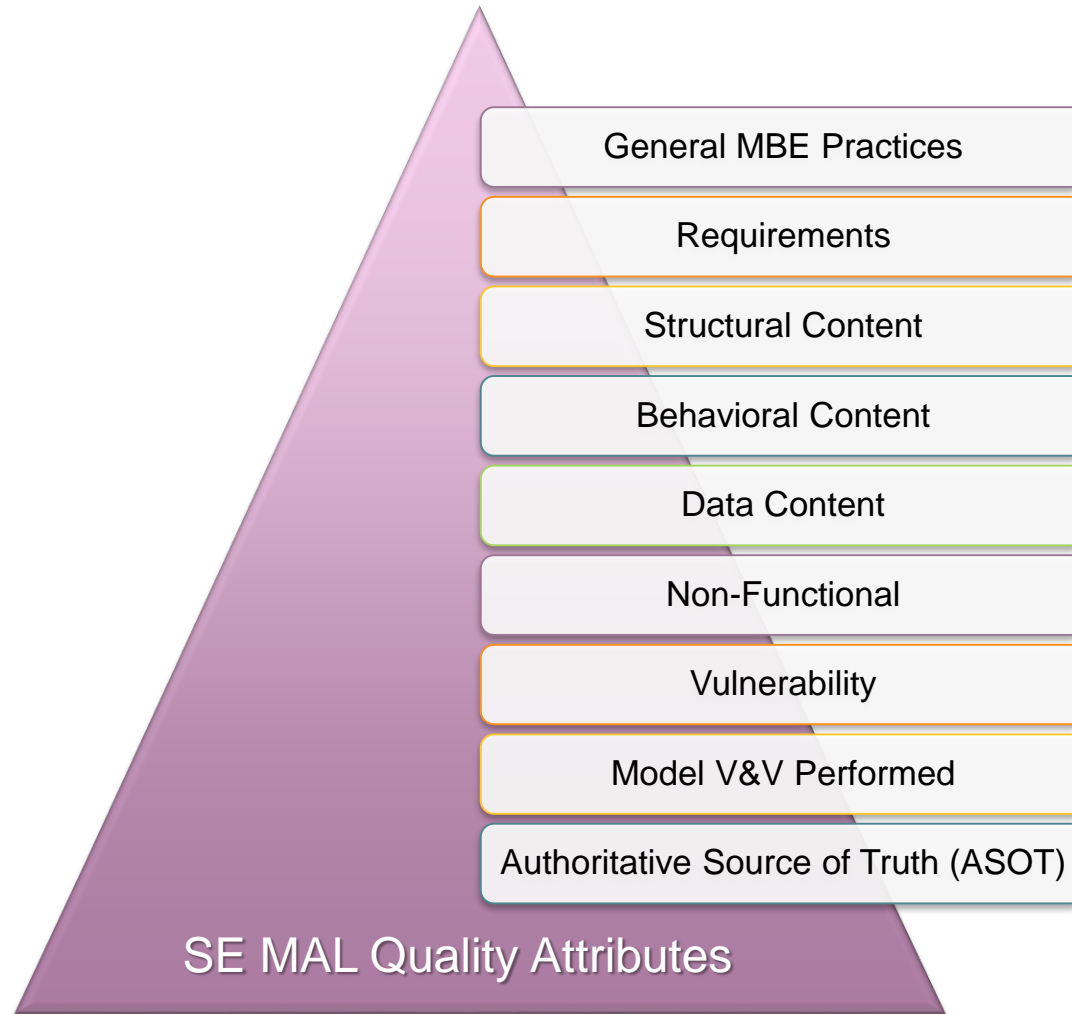
- The attributes that impact model quality are decomposed into detailed criteria that can be measured
  - *Detailed criteria are based on documented best practices, publications, and engineering judgement*
  - *Details criteria will leverage the accumulated experience of many Subject Matter Experts (SMEs)*
- Detailed criteria are
  - *Measureable*
  - *Used during the MAL assessment*
  - *Scores will be assigned based on how well the model meets the detailed criteria*



# Software MAL Quality Attributes



# Systems Engineering MAL Quality Attributes

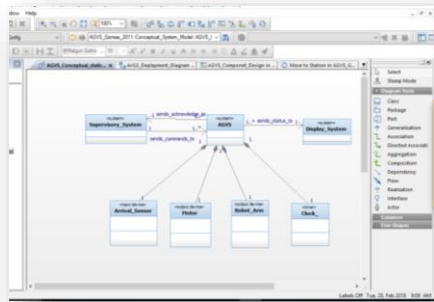


# Agenda



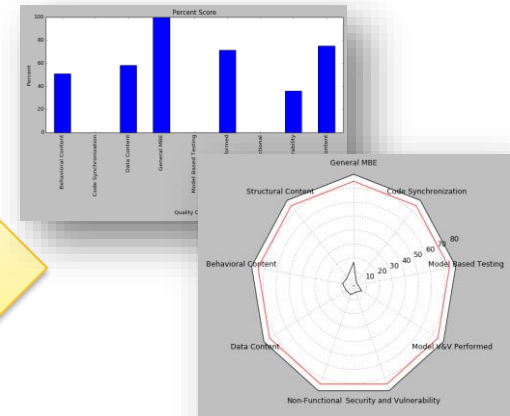
# MAL Assessment

- A MAL Assessment uses an defined and repeatable process to determine a MAL Score
  - *Quick turn around assessments*
  - *Use the model with minimal other documentation*



***Input: Software or SysML model***

**MAL Assessment**



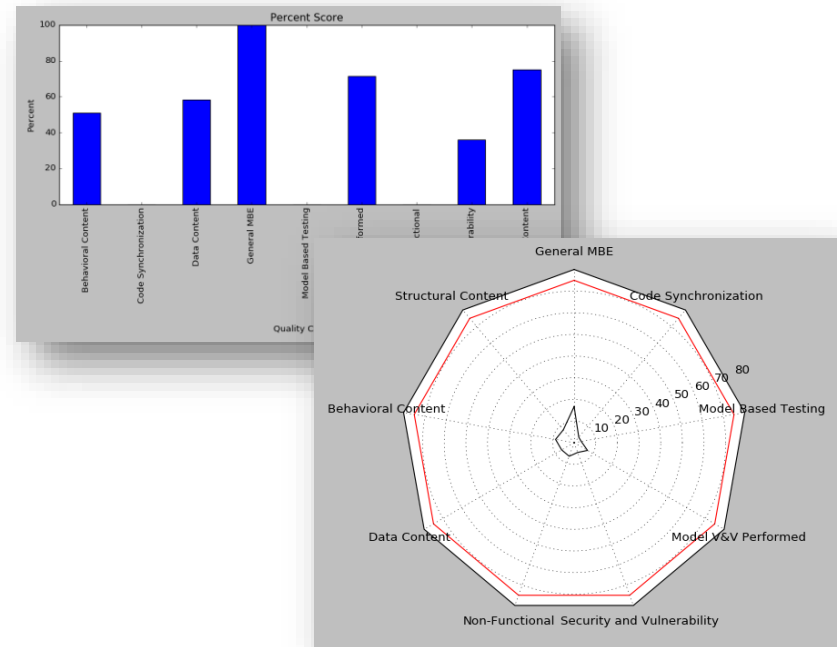
***Output: MAL Score and graphs to illustrate score***

# MAL Tool Support



- Software MAL Evaluator Tool
  - *Computes raw detailed criteria scores directly from native Rhapsody model format*
- MAL Software Scorer Tool
  - *Automatically computes MAL score, for overall software and broken down by subsystem, based on manual inputs of MAL detailed criteria*
  - *Automatically created graphs to show scores in different quality areas*
  - *Generates detailed scoring breakdown to clarify and justify scoring*
- Enterprise MAL Tool Support
  - *Initial development started*

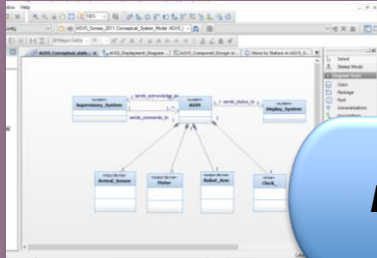
MAL Characteristic	Detailed Criteria	Weight	Percent
MBE Foundation	Established model language is used (i.e. standard modeling languages, formal DSL, etc.)	1.5	0
MBE Foundation	Modeling tool is used (versus drawing tool or power point)	1.5	0
MBE Foundation	Standards compliant modeling tool is used (e.g. Magic Draw, Rhapsody, not EA)	1	0



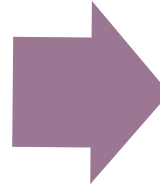
# SW MAL Assessment Process – Rhapsody Model



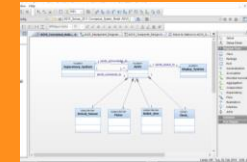
## Partial Automated Evaluation



**MAL  
Evaluator  
Tool**



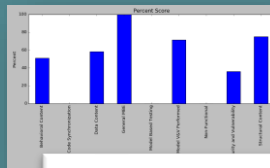
## Partial Manual Evaluation



MAL Characteristic	Detailed Criteria	Weight	Percent
1	Scenarios are modeled or listed	1	100
2	Scenarios descriptions are documented in model	1	100
3	Scenarios contain alternatives and error conditions	1	100
4	Subsystem interaction is captured in dynamic model	1	100
5	Classes with state based behavior (e.g. control classes) are in the dynamic model	1	100
6	Any design assumptions or design decisions about dynamic model are captured in model	1	100
7	Grouping system is documented	0.25	100
8	Software classes in scenarios are also defined in the static model	1.5	100
9	Classes are modeled only once in tool and reused across dynamic	1	100
10	Classes are modeled only once in tool and reused across dynamic	1	100
11	Classes are modeled only once in tool and reused across dynamic	1	100
12	Classes are modeled only once in tool and reused across dynamic	1	100
13	Classes are modeled only once in tool and reused across dynamic	1	100
14	Classes are modeled only once in tool and reused across dynamic	1	100
15	Classes are modeled only once in tool and reused across dynamic	1	100
16	Classes are modeled only once in tool and reused across dynamic	1	100
17	Classes are modeled only once in tool and reused across dynamic	1	100
18	Classes are modeled only once in tool and reused across dynamic	1	100
19	Classes are modeled only once in tool and reused across dynamic	1	100
20	Classes are modeled only once in tool and reused across dynamic	1	100



## Present Results



Category	Score	Weight	Percent
General MBE	100	0.1	1
Structural Content	55.88235294	0.15	8.38235294
Behavioral Content	62.87581699	0.15	9.43137254
Data Content	0	0.1	0
Non-Functional	0	0.1	0
Security and Vulnerability	0	0.1	0
V&V	22.22222222	0.1	2.22222222
MBT	0	0.1	0
Implementation Synchronization	0	0.1	0



## Compute Score

MAL Characteristic	Detailed Criteria	Weight	Percent
1	Scenarios are modeled or listed	1	100
2	Scenarios descriptions are documented in model	1	100
3	Scenarios contain alternatives and error conditions	1	100
4	Subsystem interaction is captured in dynamic model	1	100
5	Classes with state based behavior (e.g. control classes) are in the dynamic model	1	100
6	Any design assumptions or design decisions about dynamic model are captured in model	1	100
7	Grouping system is documented	0.25	100
8	Software classes in scenarios are also defined in the static model	1.5	100
9	Classes are modeled only once in tool and reused across dynamic	1	100
10	Classes are modeled only once in tool and reused across dynamic	1	100
11	Classes are modeled only once in tool and reused across dynamic	1	100
12	Classes are modeled only once in tool and reused across dynamic	1	100
13	Classes are modeled only once in tool and reused across dynamic	1	100
14	Classes are modeled only once in tool and reused across dynamic	1	100
15	Classes are modeled only once in tool and reused across dynamic	1	100
16	Classes are modeled only once in tool and reused across dynamic	1	100
17	Classes are modeled only once in tool and reused across dynamic	1	100
18	Classes are modeled only once in tool and reused across dynamic	1	100
19	Classes are modeled only once in tool and reused across dynamic	1	100
20	Classes are modeled only once in tool and reused across dynamic	1	100

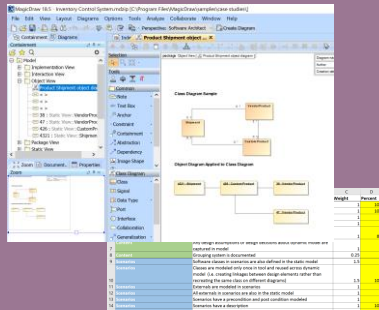
**MAL Scorer  
Tool**



# Other Model Format MAL Assessment Process



## Manual Evaluation



## Compute Score

MAL Characteristic	Detailed Criteria	Weight	Percent
1	Scenario descriptions are documented in model	1	100
2	Scenarios contain alternatives and error conditions	1	100
3	Subsystem interaction is captured in dynamic model	1	100
4	Classes with state based behavior (e.g. control classes) are in the dynamic model	1	80
5	Any design assumptions or design decisions about dynamic model are captured in model	1	100
6	Grouping system is documented	0.25	100
7	Software classes in scenarios are also defined in the static model	1.5	100
8	Classes are modeled only once in tool and reused across dynamic	1.5	100
9	Scenarios are also in the static model	1	100
10	Condition and post condition modeled	1	100
11	System	1	100

**MAL Scorer Tool**



**Note: Cameo Interop would enable us to support Magic Draw and System Architect formats by importing them into Rhapsody.**

## Present Results



General MBE	100	0.1	1
Structural Content	55.88235294	0.15	8.38235294
Behavioral Content	62.87581699	0.15	9.43137254
Data Content	0	0.1	
Non-Functional	0	0.1	
Security and Vulnerability	0	0.1	
V&V	22.22222222	0.1	2.22222222
MBT	0	0.1	
Implementation Synchronization	0	0.1	

# Agenda



# Customer Use Cases for MALs



## Monitor Progress

Perform multiple MAL Assessments thought lifecycle  
Compare the improvements between each MAL assessment (i.e. is the model growing in depth or breath)

# Customer Use Cases for MALs



Perform one or more MAL assessments to understand current model risk  
Develop strategy for mitigating risks



Perform multiple MAL Assessments thought lifecycle  
Compare the improvements between each MAL assessment (i.e. is the model growing in depth or breath)



# Customer Use Cases for MALs



# Summary





**ANY QUESTIONS?**





# References



## 1. NASA's TRL Website

[https://www.nasa.gov/directorates/heo/scan/engineering/technology/txt\\_accordion1.html](https://www.nasa.gov/directorates/heo/scan/engineering/technology/txt_accordion1.html)

- **Published Conference Papers on Software MALs**

- *J. Fant, R. Pettit, and D. Gayek “A Quantitative Approach for Calculating Model Assurance Levels” Proceedings of the 22nd IEEE Internal Symposium on Real-Time Computing (ISORC 2019).*
- *J. Fant and R. Pettit. “Model Assurance Levels (MALs) for Managing Model-Based Engineering (MBE) Development Efforts”, Proceedings of the 7th International Conference on Model Driven Engineering and Software Development (MODELSWARD) 2019.*